AFRL-RI-RS-TR-2017-243

STATISTICAL RELATIONAL LEARNING AND SCRIPT INDUCTION FOR TEXTUAL INFERENCE

UNIVERSITY OF TEXAS-AUSTIN

DECEMBER 2017

FINAL TECHNICAL REPORT

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED

STINFO COPY

AIR FORCE RESEARCH LABORATORY
INFORMATION DIRECTORATE

■ **AIR FORCE MATERIEL COMMAND** ■ **UNITED STATES AIR FORCE** ■ **ROME, NY 13441**

NOTICE AND SIGNATURE PAGE

Using Government drawings, specifications, or other data included in this document for any purpose other than Government procurement does not in any way obligate the U.S. Government. The fact that the Government formulated or supplied the drawings, specifications, or other data does not license the holder or any other person or corporation; or convey any rights or permission to manufacture, use, or sell any patented invention that may relate to them.

This report is the result of contracted fundamental research deemed exempt from public affairs security and policy review in accordance with SAF/AQR memorandum dated 10 Dec 08 and AFRL/CA policy clarification memorandum dated 16 Jan 09. This report is available to the general public, including foreign nations. Copies may be obtained from the Defense Technical Information Center (DTIC) (http://www.dtic.mil).

AFRL-RI-RS-TR-2017-243 HAS BEEN REVIEWED AND IS APPROVED FOR PUBLICATION IN ACCORDANCE WITH ASSIGNED DISTRIBUTION STATEMENT.

FOR THE CHIEF ENGINEER:

/ S /
JAMES M. NAGY
Work Unit Manager

/ S /
MICHAEL J. WESSING
Deputy Chief, Information Intelligence
Systems and Analysis Division
Information Directorate

This report is published in the interest of scientific and technical information exchange, and its publication does not constitute the Government's approval or disapproval of its ideas or findings.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From - To)*
DECEMBER 2017	FINAL TECHNICAL REPORT	NOV 2012 – MAY 2017

4. TITLE AND SUBTITLE

STATISTICAL RELATIONAL LEARNING AND SCRIPT INDUCTION FOR TEXTUAL INFERENCE

5a. CONTRACT NUMBER
N/A

5b. GRANT NUMBER
FA8750-13-2-0026

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Raymond Mooney

5d. PROJECT NUMBER
DEFT

5e. TASK NUMBER
1214

5f. WORK UNIT NUMBER
R0W4

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

University of Texas-Austin
305 E 23rd St
Austin, TX 78712

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Air Force Research Laboratory/RIED
525 Brooks Road
Rome NY 13441-4505

10. SPONSOR/MONITOR'S ACRONYM(S)
AFRL/RI

11. SPONSOR/MONITOR'S REPORT NUMBER
AFRL-RI-RS-TR-2017-243

12. DISTRIBUTION AVAILABILITY STATEMENT

Approved for Public Release; Distribution Unlimited. This report is the result of contracted fundamental research deemed exempt from public affairs security and policy review in accordance with SAF/AQR memorandum dated 10 Dec 08 and AFRL/CA policy clarification memorandum dated 16 Jan 09

13. SUPPLEMENTARY NOTES

14. ABSTRACT

Deep, logic-based approaches and statistical, weighted approaches to understanding natural-language text are often viewed as alternatives. However, they are complementary in their strengths. Logic-based approaches can draw inferences from complex, nested sentences. Statistical approaches can judge semantic similarity, and can learn highly useful regularities from large amounts of data – including inference rules encoding probabilistic common-sense knowledge. In this project, we have advanced statistical methods for learning common-sense knowledge and for identifying entity relations in text, and we have integrated logical and statistical methods to induce and effectively utilize probabilistic knowledge for appropriate, accurate inferences when comprehending documents. We have developed four different algorithmic components for aiding relation extraction and textual inference – Distributed Markov Logic Semantics, Learning Bayesian Logic Programs for Textual Inference; Stacking for Relational Extraction and Statistical Script Induction.

15. SUBJECT TERMS

Statistical Relational Learning, Textual inference, Script induction, markov logic semantics, inducing rules, statistical script induction

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			**JAMES M. NAGY**
U	U	U	UU	27	19b. TELEPHONE NUMBER (*Include area code*) **N/A**

TABLE of CONTENTS

TABLE of FIGURES

TABLE of TABLES

1 SUMMARY

Deep, logic-based approaches and statistical, weighted approaches to understanding natural-language text are often viewed as alternatives. However, they are complementary in their strengths. Logic-based approaches can draw inferences from complex, nested sentences. Statistical approaches can judge semantic similarity, and can learn highly useful regularities from large amounts of data – including inference rules encoding probabilistic common-sense knowledge. In this project, we have advanced statistical methods for learning common-sense knowledge and for identifying entity relations in text, and we have integrated logical and statistical methods to induce and effectively utilize probabilistic knowledge for appropriate, accurate inferences when comprehending documents. We have developed four different algorithmic components for aiding relation extraction and textual inference.

1. **Distributional Markov Logic Semantics:** This component developed methods for integrating *distributional lexical semantics* with logic to produce a flexible but powerful representation of sentence meaning. We designed highly effective methods for predicting *lexical entailment,* inference at the word and phrase level, as well as *lexical substitution,* context-specific paraphrasing at the word and phase level, from distributional models. We encoded the predictions as probabilistic rules to integrate them with logic-based representations, and performed probabilistic inference using *statistical relational learning (SRL).* The resulting system achieved state of the art results on a data set for the task of *recognizing textual entailment (RTE)* and also showed good results on the task of predicting *semantic textual similarity (STS).* This component has led to significant advances in the accuracy of lexical entailment predictions, the efficiency of SRL systems for textual inference, and the logic-based encoding of semantics in a way that is appropriate for probabilistic inference.
2. **Learning Bayesian Logic Programs for Textual Inference**: This component learned rules encoding probabilistic, implicit knowledge for *Knowledge Base Population (KBP*). Like the first component, it used SRL, in this case to learn *Bayesian Logic Programs* (BLPs) from a body of previously extracted facts, and then used the learned BLP to infer additional plausible relations from extracted facts when processing new documents.
3. **Stacking for Relation Extraction**: This component improved on relation extraction for Knowledge Base Population by using *stacking,* an ensemble learning technique based on training a meta-classifier. With this technique we achieved large gains over existing best systems. Importantly, we introduced *SWAF, stacking with additional features*.
4. **Statistical Script Induction**: This component learned *scripts*, stereotypical sequences of actions and events, from large text corpora, and used these scripts to infer plausible missing events. It introduced new methods that learned correlations between participants in a script, and methods that used recurrent networks to generalize over observed sequences for overall significantly improved prediction of actions and events. We also introduced a new evaluation framework using crowdsourcing that greatly improved the interpretability of evaluations.

2 INTRODUCTION

Deep understanding of natural-language text requires making inferences. Human readers naturally use commonsense knowledge to "read between the lines" and infer additional information from the explicitly stated facts. Additionally, answering many queries can require such inference. Consider the text "Barack Obama is the president of the United States." Given the query "Barack Obama is a citizen of what country?", standard *information extraction* (IE) systems cannot identify the answer since citizenship is not explicitly stated. However, a human reader possesses the commonsense knowledge that the president of a country is almost always a citizen of that country, and easily infers the correct answer.

Many types of knowledge are required to make such appropriate inferences when comprehending text. The traditional approach to inferring implicit information involves using commonsense knowledge in the form of logical rules to deduce additional information from explicitly stated facts. However, manually developing such rules is difficult and arduous, and such knowledge is probabilistic in nature rather than strictly logical. Consequently, in this project we have integrated logical and statistical methods to automatically acquire such probabilistic common-sense knowledge directly from text, and then effectively utilized it to make appropriate, accurate inferences when comprehending documents. We generally exploited methods in *statistical relational learning* (SRL) which effectively combine the strengths of symbolic, relational knowledge representation and inference together with the abilities of *probabilistic graphical models* to learn and make useful uncertain inferences in the presence of noisy data and imperfect knowledge.

A wide range of language technology applications can be cast in terms of loose implication: Given two text snippets t and h (*text* and *hypothesis*), would a human, after reading t, also consider h to be most likely true? Our approach to this task of *recognizing textual entailmet (RTE)* combined deep, logic-based and shallow, distributional methods in a novel and principled way.

First-order logic provides a powerful and flexible mechanism for representing natural language semantics. Logical inference seems predisposed to addressing textual entailment task, as it already has a notion of entailment. However, logical entailment is too strict, and misses most cases of textual entailment. On the other hand, distributional models are widely used for determining semantic similarity between words, phrases, or multi-word predicates. But in a distributional framework, there is not currently any method for handling the influence of important phenomena like negation or modal verbs on entailment. In this project, we have modeled the meaning of text through a joint representation integrating logical form with a distributional model. Distributional information is used to project uncertain, weighted inference rules into logical form, yielding a collection of unweighted and weighted clauses. We recast first-order semantics into the probabilistic models that are part of Statistical Relational AI, using *Markov Logic Networks (MLNs)* to successfully perform inferences that take advantage of logical concepts such as negation as well as weighted information on word meaning.

In addition to integrating logic and statistics for textual inference, this project has also advanced the state of the art in purely statistical models for textual inference. We have developed models for relation extraction that are based on stacking, an ensemble learning technique that trains a meta-classifier to integrate base classifier predictions. We have developed more expressive models for representing script knowledge, knowledge about event and action sequences. And we have introduced new models for lexical entailment and lexical substitution.

3 METHODS, ASSUMPTIONS, AND PROCEDURES

3.1 Distributional Markov Logic Semantics

The aim of this part of the project has been to do inferences over deep representations of sentence meaning in probabilistic logical form. Uncertain, distributional information is added as weighted inference rules, and inference is performed using Statistical Relational Learning (SRL) methods.

Weighted inferences at the word and phrase level were added using the idea that if phrase p1 lexically entails phrase p2 with a weight w, then this information can be turned into an inference rule that allows the system to infer p2 from p1 with a weight f(w) that is a function of w. The lexical entailment information can come from a resource, such as WordNet or the Paraphrase database PPDB, or from a classifier that uses distributional representations to predict lexical entailment. The function f transforming weights was learned by an SRL system on a per-resource basis.

We tested two different SRL methods, Markov Logic Networks (MLNs) and Probabilistic Soft Logic (PSL). MLNs build on undirected graphical models to implement a probabilistic logic based on a probability distribution over worlds; PSL is a truth-functional approach that computes more directly with weighted rules.

We tested our approach on three tasks: Recognizing Textual Entailment (RTE), Semantic Textual Similarity (STS), and question answering.

At the distributional level, we trained models to predict lexical entailment, the degree to which one word or phrase H entails another word or phrase w. Multiple features that can be extracted from distributional data are relevant to this task: the overall contextual similarity of H and w, the degree to which H seems to be a word high up in the taxonomy, and the degree to which H appears in more contexts than w. Importantly we found that a simple model that seemed to just memorize typical hypernyms was instead learning to detect words high up in the taxonomy by learning to recognize distributional contexts that constitute Hearst patterns, a classical pattern-based approach to hypernymy detection.
We also developed models to predict lexical substitution, paraphrasing specific to the sentence context, focusing on simple models that integrate fit of the potential paraphrase with both the target word and the context. We also developed models to predict entity properties (taxonomic as well as perceptual and functional properties from a given resource) from distributional vectors.
To be able to better handle unknown words, we developed models for predicting properties of words (taxonomic, perceptual, functional, and social properties) from distributional data. We experimented with a wide variety of models, including a new method based on label propagation, in particular modified adsorption.

3.2 Inference Rule Learning and Relation Extraction

The aim of this part of the project was to automatically learn Bayesian Logic Programs (BLPs) from text-extracted information and use the resulting probabilistic model to make

accurate inferences from facts extracted from future documents. We experimented with learning from facts extracted from corpora, and facts extracted from DBPedia. The aim was to infer plausible inference rules that went beyond explicitly stated knowledge to read between the lines. Evalution was on the KBP slot filling task.

3.3 Stacking for Relation Extraction

In 2015, we replaced our work on Bayesian Logic Programming, with a new project on ensembling for relation extraction that directly addressed the slot filling task of the Knowledge Base Population (KBP) challenge. In this project, we used stacking, an ensemble learning approach which trains a final meta-classifier to optimally combine the results of multiple systems is a very general and effective approach. First, we added as input to the meta-classifier, a new feature that uses the document provided for each KBP query that disambiguates the entity in the query by providing a document in which the intended entity is mentioned. Our new feature computes the standard TF-IDF-weighted cosine similarity between the provenance passage for a given query result and this query-disambiguating document. The intuition behind this feature is that, if the answer does involve the correct query entity, then this document similarity should be reasonably high. By providing the stacker with this additional information, it should be able to better decide if the answer is indeed correct.

Stacking generally takes as input the classification result and its corresponding certainty for each system in the ensemble. However, KBP systems must also provide provenance information; each extracted slot-filler must include a pointer to a document passage that supports it. Therefore, we also explored enhancing our stacking approach by including additional input features that quantify how much the ensembled system agree on provenance.

An important limitation of our initial stacking approach was that it required supervised training data for each system, which provides evaluated results on their performance for a shared set of queries. We had been using performance data for systems in the KBP competition in year N to provide training data for training a stacker that combines them for the competition in year N+1. Although systems change from year to year, we found that their performance data from the previous year still provided effective training data for the stacker. However, this only allowed ensembling systems for which we had historical performance data, which is not true for all systems. Therefore, the stacker could not take advantage of data from new systems for which we had no evaluated prior results. Therefore, we developed a way to include such systems using an “unsupervised” approach and combining this with supervised training for systems for which we do have such data. We first combined the outputs for all systems for which we did not have historical data using the unsupervised approach to ensembling developed by the JHU team for the 2013 KBP Filtering task. Next, we ensembled its output with the output of the supervised stacking approach using a simple approach that combined fillers for list-valued slots and selected the most confident answer for single-valued slots. By combining evidence from all systems, using historical performance data where available, this approach combines supervised and unsupervised ensembling to exploit the advantages of both.

We generalized our approach of Stacking With Additional Features (SWAF) to problems beyond KBP, in particular computer vision; particularly, we applied it to the ImageNet object detection challenge. Like the KBP tasks, this task requires systems to provide "provenance" for the objects they detect in an image in the form of a bounding box around each detected object. Analogous to its use in KPB ensembling, we used this provenance to create auxiliary features which capture the overlap in bounding boxes, thereby measuring provenance agreement.

3.4 Statistical Script Induction

The aim of this part of the project was to automatically induce "scripts", i.e. knowledge of stereotypical sequences of actions, from large corpora in order to improve the understanding of text. Previous approaches to statistical script induction (originated by Chambers and Jurafsky) employ an impoverished representation of events that only includes a verb and a single dependent entity. We developed a more complex event representation for use in statistical script models, capable of directly capturing interactions between multiple entities. As with previous script induction approaches, we first used Stanford NLP tools to dependency-parse documents and perform co-reference in order to identify entity mentions. However, instead of extracting verb-entity pairs from the resulting pre-processed documents, we extracted events of the form $v(x,y,z)$, where v is a verb, x is its subject, y is its direct object, and z is its prepositional object. We then construct a statistical model of co-occurring events by estimating the probability for $P(e_1 \mid e_2)$, which gives the probability that event e_1 occurs after e_2 in a document while capturing the mapping between the three syntactic dependents of e_1 and those of e_2. For example, the model can learn that there is a relatively high probability that "*x* gives *y* to z" after "z orders *y* from *x*".

A second model we developed was based on deep recurrent neural network methods using Long Short Term Memory (LSTM). LSTMs incorporate explicitly controllable memory units that allow them to learn long-range temporal dependencies, which are very difficult to learn using traditional recurrent networks. Several researchers have also recently used LSTMs to successfully translate French to English and generate natural-language descriptions of images and videos. As before, we used the Stanford dependency parser and co-reference resolver to produce a sequence of verb-argument events. Given this sequence, using the CAFFE software package from the University of California Berkeley, we trained an LSTM recurrent network to predict the next event after processing each event. The network learns a distributed hidden state representation that allows it to effectively help predict subsequent events.

Our third model was motivated by the work on the "Skip Thought" approach to producing distributed sentence representations by training a pair of encoder-decoder LSTMs to predict the context sentences around the target sentence, we developed a new approach that does not rely on linguistic preprocessing. Our third model used LSTMs trained as sentence-level language models which try to directly predict the sequence of words in the next sentence from a learned representation of the previous sentence using no linguistic preprocessing.

Following previous work, we evaluated our approaches using the "narrative cloze" task, in which a random event in a document is removed and the ability of the model to accurately

infer the missing event is measured. The Narrative Cloze evaluation is fully automated; however, its results are not easily interpretable, and it is not clear if they correlate with human judgments. Therefore, we developed and performed a new evaluation of script learning that uses crowdsourced human assessments to directly test the event inferences of four previously published script-learning methods, including our own. Additionally, we tested whether our script prediction models would be useful for coreference resolution and for the related task of implicit role prediction. Implicit role prediction is the identification of a predicate's argument that is not a syntactic dependent of the predicate but is present in the larger discourse context.

4 RESULTS AND DISCUSSION

4.1 Distributional Markov Logic Semantics

We scaled up our system to be able to handle naturally occurring sentences, and evaluated it on two existing shared tasks: Textual Entailment (RTE) and Semantic Textual Similarity (STS). Two main innovations have enabled us to do this. The first was that we used distributional similarity not only at the word level but also at the short phrase level, deriving rules

assessing for example the similarity of "ketchup" and "tomato sauce". This change led to a sizable performance improvement on both tasks. In contrast to existing approaches that compile large collections of textual inference rules, we use distributional similarity of phrases to generate inference rules "on the fly". The second main innovation was that we used more flexible probabilistic combinations of evidence in order to compute degrees of sentence similarity for

STS and to help compensate for parser errors. We replace deterministic conjunction by an average combiner, which encodes causal independence. Our framework was the first to handle both RTE and STS in a single system, and achieved reasonable results on both tasks. This work is described in Beltagy et al. (2013).

The size of Markov Networks constructed during inference in Markov Logic Networks is a main source for the performance issues of the probabilistic inference. We have explored Probabilistic Soft Logic (PSL) as an alternative probabilistic inference framework. In contrast to MLNs, PSL has an efficient solving procedure. However, PSL is truth-functional, which can lead to it drawing the wrong inferences. So we use it for the task of predicting sentence similarity (STS) and sentence paraphrasing, but not Textual Entailment, where deeper inferences are required. As the formula for conjunction implemented in PSL is too strict for our purposes, we replaced it by a weighted average. Inference with PSL was not only much faster, it also yielded better results on both the STS video dataset and the Microsoft paraphrase corpus. On STS, correlation (Pearson) was 0.73 for MLN and 0.8 for PSL (the current leading approach achieves a correlation of 0.87), and on the Microsoft paraphrase corpus, which is much more complex, correlation increased from 0.25 to 0.5 through the use of PSL. This work is described in Beltagy, Erk and Mooney (2014).

In March 2014, we participated in Task 1 of SemEval, the Semantic Evaluation workshop: “Evaluation of compositional distributional semantic models on full sentences through semantic relatedness and entailment”. The task involved both RTE and STS subtasks on the SICK dataset (Sentences Involving Compositional Knowledge). We obtained an accuracy of 73% on RTE, and a Pearson correlation of 0.71 on STS, as reported in Beltagy et al. (2014).

Markov Logic Networks can handle all of first-order logic, and have a principled basis in probabilistic logic. However, the networks can grow very large, leading to performance issues. We integrated a new inference algorithm based on SampleSearch into Alchemy (the MLN inference system that we are using) to improve run time. We also introduced a modified closed-world assumption that significantly reduces the size of the ground network, thereby making inference feasible. This step has the added benefit of removing impossible

literals from the system, thereby making inference more accurate. Evaluation on the training portion of the SICK RTE data yielded an accuracy of 71.8% for the modified system (original system: 56.9%) with an average runtime of 7s per datapoint (original system: 2min 27s). These results are reported in Beltagy and Mooney (2014).

As reported above, we addressed problems resulting from the fact that MLN (and most probabilistic logic frameworks) work in finite domains, and make the domain closure assumption, and also problems resulting from the fact that we make the closed-world assumption to substantially reduce the size of the graphical models constructed by the MLNs. We evaluated these representational enhancements on a synthetic RTE dataset of different quantifiers and monotonicity directions and confirm that our system achieves 100% accuracy on it, as reported in Beltagy and Erk (2014).

We developed a supervised classifier that uses distributional vector representations of two words A, B to decide whether A is a hypernym of B. The classifier showed good results, with an accuracy of 84% on a four-way distinction of semantic relations (hypernymy, co-hyponymy, meronymy, and random) on the BLESS dataset and similar results on the Entailment dataset. In contrast to preliminary results listed in the previous report, our approach significantly outperforms a previous system by Baroni et al. Our classifier allows for an intuitive interpretation as performing a selection of features on which feature inclusion holds between hypernym and hyponym. This result is reported in Roller, Erk and Boleda (2014).

In subsequent experiments with models for lexical entailment, we found that a simple model that seemed to just memorize typical hypernyms was instead learning to detect words high up in the taxonomy by learning to recognize distributional contexts that constitute Hearst patterns, a classical pattern-based approach to hypernymy detection. By incorporating these contexts as features into a lexical entailment classifier we were able to obtain state-of-the-art performance on lexical entailment. These results were reported in Roller and Erk (2016).

Beltagy et al. (2016) is a journal article that summarizes the architecture developed in this part of the project. In this article, we also report a new state-of-the art performance on the SICK dataset for Textual Entailment (RTE). Our accuracy is 85.06, while the previous state of the art was 84.58. The organizers of the textual entailment task at SemEval 2014 that introduced SICK concluded that purely compositional approaches seemed to do worse on the task than non-compositional ones; with this new paper we show that it is possible for a model that performs deep compositional semantic analysis to get state-of-the-art performance on this task. The system that we describe in that paper includes a new supervised lexical and phrasal entailment classifier, which played an important role in the success of the system. Without it, the accuracy was 80.37 instead of 85.06. A lexical/phrasal entailment classifier predicts whether a sentence involving p1 entails the same sentence where p1 is replaced by p2. For example, “street” entails “road”, and “hole” entails “earth” in the context of “digging a hole/digging the earth”. Figure 1 illustrates the overall architecture of the system.

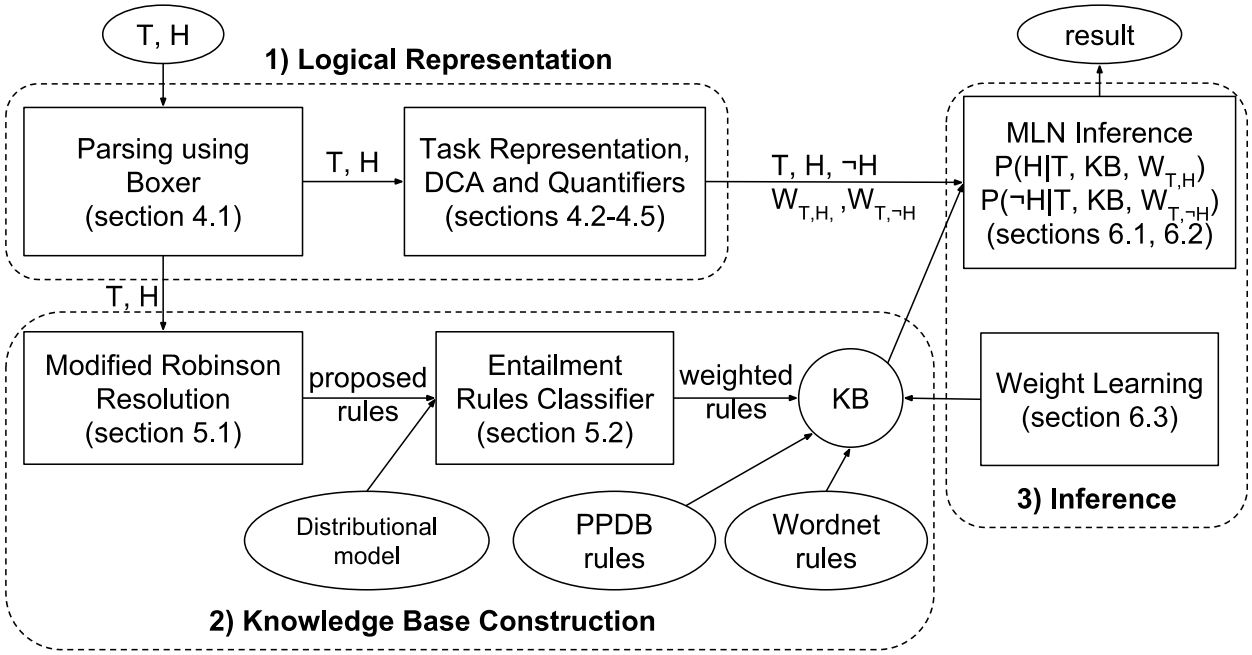

Figure 1: Distributional Markov Logic Semantics system architecture

The use of such a lexical/phrasal classifier was made possible by a logic-based alignment system that, given a Text/Hypothesis pair, removed the parts of the sentences that they shared using a modified version of Robinson Resolution, to identify the phrases that need to be entailed in order for the Text to entail the Hypothesis. Running the modified Robinson Resolution on the training part of the SICK RTE dataset yielded training data for the lexical/phrasal entailment classifier, and running the modified Robinson Resolution on the SICK test sentences yielded the test data for the classifier.

An evaluation of the lexical/phrasal entailment classifier on its own yielded an accuracy of 83.0 on the SICK test set. An in-depth evaluation of the classifier showed that both WordNet information and distributional information contributed to the success of the classifier.

We also evaluated a compositional distributional approach on the task of phrasal entailment. We found that this approach is effective at flagging phrase pairs that are *not* entailing, for example because of prepositions that change sentence meaning or because of a difference in semantic roles (“man eats near kitten”/ “kitten eats”), but not so much at identifying entailing phrase pairs. For identifying entailing phrase pairs, an approach that integrates entailment information at the word level showed much better performance.
We also integrated information from the paraphrase collection PPDB. We used a rule-based technique to translate entries from the paraphrase collection to logical rules. The rules from PPDB and from the lexical/phrasal entailment classifier come with different weights. We used weight learning to map these weights to MLN weights. We learned one weights scaling factor per rules source. We use simple grid-search to learn the scaling factors.
Our approach to lexical substitution, context-specific paraphrasing, was found to perform close to state-of-the-art on the task of ranking a set of given paraphrases, while being much simpler than the state-of-the-art system. On the task of selecting paraphrases from the whole

vocabulary, our model was found to have superior performance. We noted that our model particularly benefitted from taking into account the frequency of paraphrase candidates, in that it tended to prefer frequent words, while other systems often suggested misspelled paraphrases. These results are reported in Roller and Erk (2016).
We tested our approaches to the prediction of properties for unknown words from their distributional representations on two datasets from psychology along with WordNet hypernyms and the General Inquirer. We found that it is possible to use the same methods to learn properties from the psychology datasets as well as WordNet hypernyms and General Inquirer properties. We found our new method based on modified adsorption to outperform other methods across all datasets, beating previously used methods. A paper on this work will be submitted to the journal of Natural Language Engineering at the end of this year.

4.2 Learning Bayesian Logic Programs for Textual Inference

An on-line rule learning algorithm that we developed for inducing the first-order

rules for a BLP from noisy extractions was more than 100 times faster than our previous approach while maintaining a similar quality of the induced rules. This was demonstrated by evaluating the approach on the IC corpus from the DARPA Machine Reading Program (Raghavan et al., 2013).
We participated in the 2013 NIST KBP (Knowledge-Based Population) slot-filling task by using a BLP developed for the KBP ontology to make inferences from text extractions with the goal of increasing recall (Bentor et al., 2013). We used the publicly distributed version of the CUNY BLENDER system as the base-level KBP extractor. During testing, we used a learned BLP to infer additional facts from the facts extracted by BLENDER, and submitted two sets of results for the competition, one with inferred relations added as well as a baseline set of results without BLP inferences.

In order to assemble a large training set for learning a BLP appropriate for KBP, we mapped 26 of the 41 predicates in the KBP ontology to relations in the open-linked database, DBPedia. We then used our previously developed on-line BLP rule learner to learn a BLP from 912,375 mapped facts from DBpedia. For example, one learned rule was: “If person B is a key employee of organization A, then B is probably a shareholder in A.” We also added a set of manually written inference rules to the rules learned from DBPedia. Given the poor performance of using EM (Expectation Maximization) to learn parameters for BLP models, we developed a simpler estimator for the noisy-or parameters in our BLPs by independently computing the accuracy of each rule on the training data.

Unfortunately, partly because the KBP evaluation is focused on evaluating the extraction of explicitly-stated facts rather than probable inferences, the BLP inferences failed to improve recall and actually resulted in an overall decrease in F-measure (form 0.123 to 0.108), as reported in Bentor et al. (2014). In our officially submitted results, we preferred inferred slot fillers to explicitly extracted ones in order to emphasize the role of inference. Subsequent to the official evaluation, we conducted an additional experiment in which we preferred inferred fillers to extracted ones *only* if their estimated confidence was higher. This version generated 7 additional fillers that were judged to be correct, resulting in an increase in recall (from .079 to .085) with only a minor decrease in F-measure (from 0.123 to 0.121). This

result provides some evidence for the value of BLP textual inference despite the limitations of the KBP evaluation with respect to evaluating this capability.
We entered a system based on BLP-TI in the 2014 KBP English Slot Filling task, held in July 2014. We submitted 4 runs of our system, including a baseline run based on the top-performing 2013 KBP English Slot Filling System of Roth et al., LSV, and several runs that experimented with inference techniques using BLPs and thresholding strategies. As expected, inference did improve the system's recall, beating the baseline extractor by approximately 1%, but we were not able to outperform the baseline system's F1 score. We partially attribute this to an evaluation that did not target probabilistic inferences specifically. The utaustin system, as submitted, ranked 4th in the competition, but this is due to the strength of the baseline extractor from LSV. Our results are described in Bentor et al. (2014).

4.3 Stacking for Relation Extraction

We employed stacking using a L1-regularized linear SVM to ensemble all systems that competed in both the 2013 and 2014 KBP (English Slot Filling) tracks, training on 2013 data and testing on 2014 data. The resulting ensemble outperformed all systems in the 2014 competition, obtaining an F1 of 48.6% compared to 39.5% for the best performing system in the most recent competition. By including features encoding agreement on provenance, we further improved our F1 score for the 2014 ESF task to 50.1%, Table 2 (Viswanathan et al., 2015).

Table 1: Performance of baselines on all 2014 SFV dataset (65 systems)

Baseline	Precision	Recall	F1
Union	0.067	**0.762**	0.122
Voting (threshold learned on 2013 data)	**0.641**	0.288	0.397
Voting (optimal threshold for 2014 data)	0.547	0.376	**0.445**

Table 2: Performance on the common systems dataset (10 systems) for various configurations. All approaches except the Stanford system are our implementations.

Approach	Precision	Recall	F1
Union	0.176	**0.647**	0.277
Voting (threshold learned on 2013 data)	**0.694**	0.256	0.374
Best ESF system in 2014 (Stanford)	0.585	0.298	0.395
Voting (optimal threshold for 2014 data)	0.507	0.383	0.436
Stacking	0.606	0.402	0.483
Stacking + Relation	0.607	0.406	0.486
Stacking + Provenance (document)	0.499	0.486	0.492
Stacking + Provenance (document) + Relation	0.653	0.400	0.496
Stacking + Provenance (document and offset) + Relation	0.541	0.466	**0.501**

Table 3: Results on 2015 Cold Start Slot Filling (CSSF) task using the official NIST scorer

Methodology	Precision	Recall	F1
Combined stacking and constrained optimization with auxiliary features	0.4679	**0.4314**	**0.4489**
Top ranked SFV system in 2015 (Rodriguez 2015)	0.4930	0.3910	0.4361
Stacking using BGCM instead of constrained optimization	**0.5901**	0.3021	0.3996
BGCM for combining supervised and unsupervised systems	0.4902	0.3363	0.3989
Stacking with auxiliary features described in Rajani (2017)	0.4656	0.3312	0.3871
Ensembling approach described in Viswanathan (2015)	0.5084	0.2855	0.3657
Top ranked CSSF system in 2015 (Angeli 2015)	0.3989	0.3058	0.3462
Oracle Voting baseline (3 or more systems must agree)	0.4384	0.2720	0.3357
Constrained optimization approach described in Wang (2013)	0.1712	0.3998	0.2397

Table 4: Results on 2015 Tri-lingual Entity Discovery and Linking (TEDL) using official NIST scorer and CEAF metric

Methodology	Precision	Recall	F1
Combined stacking and constrained optimization	0.686	**0.624**	**0.653**
Stacking using BGCM instead of constrained optimization	0.803	0.525	0.635
BGCM for combining supervised and unsupervised outputs	0.810	0.517	0.631
Stacking with auxiliary features described in Rajani (2017)	0.813	0.515	0.630
Ensembling approach described in Viswanathan (2015)	**0.814**	0.508	0.625
Top ranked TEDL system in 2015 (Sil 2015)	0.693	0.547	0.611
Oracle Voting baseline (4 or more systems must agree)	0.514	0.601	0.554
Constrained optimization approach	0.445	0.176	0.252

Table 5: Results on 2016 Cold Start Slot Filling (CSSF) task using the official NIST scorer

Method	Precision	Recall	F1
Stacking with both provenance + instance auxiliary features	0.258	**0.439**	**0.324**
Stacking with just provenance auxiliary features	0.252	0.377	0.302
Stacking with just instance auxiliary features	0.257	0.346	0.295
Stacking without auxiliary features	**0.311**	0.253	0.279
Top ranked CSSF system in 2016 (Zhang 2016)	0.265	0.302	0.260
Oracle Voting baseline (4 or more systems must agree)	0.191	0.379	0.206
Mixtures of Experts (ME) model	0.168	0.321	0.180

Table 6: Results on 2016 Entity Discovery and Linking (EDL) task using the official NIST scorer and the CEAFm metric

Method	Precision	Recall	F1
Stacking with both provenance + instance auxiliary features	0.739	**0.600**	**0.662**
Stacking with just provenance auxiliary features	**0.767**	0.544	0.637
Stacking with just instance auxiliary features	0.752	0.542	0.630
Stacking without auxiliary features	0.723	0.537	0.616
Top ranked EDL system in 2016 (Sil 2016)	0.717	0.517	0.601
Mixtures of Experts (ME) model	0.721	0.494	0.587
Oracle Voting baseline (4 or more systems must agree)	0.588	0.412	0.485

As discussed in the previous chapter, we extended our approach by an unsupervised ensembling step to be able to use more base systems. By combining evidence from all systems, using historical performance data where available, this approach combines supervised and unsupervised ensembling to exploit the advantages of both. Using the gold-standard answer set provided for a small set of the KBP queries during the 2015 ColdStart competition, we have shown that this approach gives the best overall performance, and was used as our primary submission to the 2015 competition.

The complete version of our Stacking with Auxiliary Features (SWAF) was used to ensemble all systems that competed in the KBP 2015 competitions, we achieved improved state-of-the-art results on *two* separate NIST KBP challenge tasks – Cold Start Slot-Filling and Tri-lingual Entity Discovery and Linking, Tables 3 and 4 respectively. Our approach outperforms the best individual system in the original competition as well as other competing ensembling methods (such as voting, unsupervised ensembling, and supervised ensembling of previously-known systems) on both tasks in the most recent 2016 competition; verifying the generality and power of our new approach to combining supervised and unsupervised ensembling. This result is described in a paper at EMNLP 2016 (Rajani and Mooney, 2016). Experiments on the 2016 competitions clearly demonstrate the value of the auxiliary features we give to the meta-classifier, allowing it to weight the contribution of individual systems based on the slot and/or entity types under consideration and its provenance agreement with other systems, Tables 5 and 6 respectively.

At IJCAI 2017, we published an overview paper on SWAF (Rajani and Mooney, 2017) that included a comprehensive set of applications and evaluations of our technique. The paper describes the general SWAF method that learns to fuse additional relevant information from multiple component systems as well as input instances to improve performance. We use two types of auxiliary features - instance features and provenance features. The instance features enable the stacker to discriminate across input instances and the provenance features enable the stacker to discriminate across component systems. When combined, our algorithm learns to rely on systems that not just agree on an output but also the provenance of this output in conjunction with the properties of the input instance. We demonstrate the success of our approach on three very different and challenging natural language and vision problems: Slot Filling, Entity Discovery and Linking, and ImageNet Object Detection. We obtain new state-of-the-art results on the first two tasks and significant improvements on the ImageNet task, thus verifying the power and generality of our approach.

We also applied SWAF (Rajani and Mooney, 2017) to Visual Question Answering (VQA), a challenging task that requires systems to jointly reason about natural language and vision in order to answer a natural language question about an image. We developed three categories of auxiliary features that can be inferred from an image-question pair: question and answer types, bag-of-words question features, and deep visual image features. Using SWAF with these auxiliary features to effectively ensemble three recent systems, we obtained a new state-of-the-art performance for VQA. We then extended this approach to include "visual explanations" as additional auxiliary features. Several existing VQA systems provide "heat maps" of the image regions that are attended to when answering a particular question. The key idea is that we trust systems' agreement on an answer more if they also agree on its explanation in the form of visual heatmaps. Results demonstrate a modest improve in VQA performance by adding these explanation-based auxiliary features. A paper discussing this result appeared at the IJCAI-17 *Workshop on Explainable AI (XAI).*

4.4 Statistical Script Induction

In Pichotta and Mooney (2014), we used the "Narrative Cloze" evaluation introduced by Chambers and Jurafsky to evaluate our statistical script model based on multi-argument events. We demonstrated that, compared to previous models, our model was able to more accurately predict events that are deleted from a document.

We also developed a new evaluation of script learning that uses crowdsourced human assessments to directly test the event inferences of four previously published script-learning methods, including our own. We found that human judgments correlate with Narrative Cloze performance; which, to our knowledge, has not been previously demonstrated. This provides justification for the Narrative Cloze with respect to comparative evaluation; however, we also found that human judgments provide more interpretable results that enable more productive error analysis.

Our second model, which uses LSTMs to predict the next event component and which uses as input passed text with coreference analysis, was shown to outperform previously developed script-learning approaches on the prediction of both entity IDs (as encoded in coreference chains) and on the prediction of head nouns of held-out entities (Pichotta and Mooney 2016a).

We evaluated our third model, which uses LSTMs as sentence-level language models to predict event sequence information, on the standard "narrative cloze" task. We found that the direct token-based approach outperformed the event-based one (Pichotta and Mooney 2016b).

In addition, we incorporated our learned script models into an extant co-reference resolution system, specifically the most recent version of the Berkeley co-reference system. We evaluated our approach on the CoNLL 2012 shared task dataset, which is based on the OntoNotes corpus. The script features provided a very modest improvement in overall accuracy (less than 1%); however, it provided some significant improvements on particularly difficult classes of co-reference decisions such as nominal/proper mentions with nominal/proper antecedents where the head nouns do not match.

We also used script models to predict event participants, in a new cloze task version of the task of predicting implicit arguments. We automatically constructed the data from the OntoNotes dataset by removing one participant that also appeared elsewhere in the discourse context. We found that event script knowledge, modeling how likely one event would co-occur with a sequence of contextual events, greatly improved prediction accuracy over baselines using only word-level knowledge. The performance was further boosted when entity salience features were added to the model. We have ongoing work on fitting the model to a much harder (and much smaller) human-labeled implicit argument task; we hope to submit a paper on this work to a conference by the end of 2017.

5 CONCLUSIONS

In this project, we have advanced statistical methods for learning common-sense knowledge and for identifying entity relations in text, and we have integrated logical and statistical methods to induce and effectively utilize probabilistic knowledge for appropriate, accurate inferences when comprehending documents. In the area of Distributional Markov Logic Semantics, we have made significant advances in the accuracy of lexical entailment predictions, the efficiency of SRL systems for textual inference, and the logic-based encoding of semantics in a way that is appropriate for probabilistic inference. In the area of relation extraction, we set aside our initial work on learning Bayesian Logic Programs for textual inference, as there is currently no good way to evaluate the resulting "reading between the lines" rules. We instead focused on stacking for relation extraction, showing that this technique achieves large gains over existing systems. Provenance-like additional features are important for helping the meta-classifier learn a good model. In the area of statistical script induction, we developed state-of-the-art systems, both based on text alone and based on a parse and coreference analysis.

6 REFERENCES

6.1 References to Papers by Our Group

6.1.1 Journal Articles

Beltagy, I., Roller, S., Cheng, P., Erk, K., and Mooney, R.J., "Representing Meaning with a Combination of Logical and Distributional Models," *Computational Linguistics*, **42**:4, 2016.

6.1.2 Conference Papers

Rajani, N.F. and Mooney, R.J., "Stacking With Auxiliary Features," *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence (IJCAI)*, Melbourne, Australia, 2017.

Rajani, N.F. and Mooney, R.J., "Combining Supervised and Unsupervised Ensembles for Knowledge Base Population," *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Austin, TX, 2016.

Pichotta, K. and Mooney, R.J., "Using Sentence-Level LSTM Language Models for Script Inference," *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL)*, Berlin, Germany, 2016.

Pichotta, K. and Mooney, R.J., "Learning Statistical Scripts With LSTM Recurrent Neural Networks," *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence (AAAI)*, Phoenix, AZ, 2016.

Viswanathan, V., Rajani N.F., Bentor, Y., and Mooney, R.J., "Stacked Ensembles of Information Extractors for Knowledge-Base Population," *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics (ACL)*, Beijing, China, 2015.

Beltagy, I., and Erk, K., "On the Proper Treatment of Quantifiers in Probabilistic Logic Semantics," *Proceedings of the International Conference on Computational Semantics*, London, UK, 2015.

Beltagy, I., Erk, K., and Mooney, R.J., "Probabilistic Soft Logic for Semantic Textual Similarity," *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (ACL*), Baltimore, MD, 2014.

Pichotta, K. and Mooney, R.J., "Statistical Script Learning with Multi-Argument Events," *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics (EACL*), Gothenburg, Sweden, 2014.

Roller, S., Erk, K., and Boleda, G. "Inclusive yet selective: supervised distributional hypernymy detection," *Proceedings of the International Conference on Computational Linguistics (CoLing)*, Dublin, Ireland, 2014.

Beltagy, I., Chau, C., Boleda, G., Garrette, D., Erk, K., and Mooney, R., "Montague Meets Markov: Deep Semantics with Probabilistic Logical Form," *Proceedings of the Second Joint Conference on Lexical and Computational Semantics (*SEM),* Atlanta, GA, 2013.

6.1.3 Workshop Papers

Rajani, N. and Mooney, R.J., "Using Explanations to Improve Ensembling of Visual Question Answering Systems," Proc*eedings of the IJCAI Workshop on Explainable AI*, Melbourne, Australia, 2017.

Rajani, N., and Mooney, R.J., "EDL 2016 UTAustin System Description," Proceedings of Text Analysis Conference (TAC), Gaithersburg, MD, 2016.

Pichotta, K. and Mooney, R.J., "Statistical Script Learning with Recurrent Neural Networks," *Proceedings of the EMNLP-2016 Workshop on Uphill Battles in Language Processing*, Austin, TX, 2016.

Rajani, N., and Mooney, R. J., "Stacked Ensembles of Information Extractors for Knowledge-Base Population by Combining Supervised and Unsupervised Approaches," *Proceedings of the Text Analysis Conference (TAC*), Gaithersburg, MD, 2015.

Bentor, Y., Viswanathan, V., and Mooney, R.J., "University of Texas at Austin KBP 2014 Slot Filling System: Bayesian Logic Programs for Textual Inference," *Proceedings of the Seventh Text Analysis Conference: Knowledge Base Population (TAC 2014*), Gaithersburg, MD, 2014.

Beltagy, I., Roller, S., Boleda, G., Erk, K., and Mooney, R., "UTexas: Natural Language Semantics using Distributional Semantics and Probabilistic Logic," *Proceedings of the International Workshop on Semantic Evaluation (SemEval),* Dublin, Ireland, 2014.

Beltagy, I. and Mooney, R.J., "Efficient Markov Logic Inference for Natural Language Semantics," *Proceedings of the AAAI-14 Workshop on Statistical Relational AI (StarAI),* Quebec City, Canada, 2014.

Beltagy, I., Erk, K., and Mooney, R.J., "Semantic Parsing using Distributional Semantics and Probabilistic Logic," *Proceedings of the ACL-14 Workshop on Semantic Parsing*, Baltimore, MD, 2014.

Bentor, Y., Harrison, A., Bhosale, S., and Mooney, R.J., "University of Texas at Austin KBP 2013 Slot Filling System: Bayesian Logic Programs for Textual Inference," *Proceedings of the Sixth Text Analysis Conference (TAC),* Washington D.C., 2013.

Raghavan, S.V. and Mooney, R.J., "Online Inference-Rule Learning from Natural-Language

Extractions," *Proceedings of the AAAI-13 Workshop on Statistical Relational AI (StarAI*), Bellevue, WA, 2013.

6.2 Other References

Gabor Angeli, Victor Zhong, Danqi Chen, Arun Chaganty, Jason Bolton, Melvin Johnson Premkumar, Panupong Pasupat, Sonal Gupta, and Christopher D. Manning, "Bootstrapped self training for knowledge base population." In *Proceedings of the Eighth Text Analysis Conference*, 2015.

Miguel Rodriguez, Sean Goldberg, and Daisy Zhe Wang, "University of Florida DSR lab system for KBP slot filler validation 2015." In *Proceedings of Text Analysis Conference*, 2015.

Avirup Sil, Georgiana Dinu, and Radu Florian, "The IBM systems for trilingual entity discovery and linking at TAC 2015." In *Proceedings of Text Analysis Confer*ence, 2015.

I-Jeng Wang, Edwina Liu, Cash Costello, and Christine Piatko, "JHUAPL TAC-KBP2013 slot filler validation system." In *Proceedings of Text Analysis Conference*, 2013.

Y. Zhang, A. Chaganty, A. Paranjape, D. Chen, J. Bolton, P. Qi, and C. Manning, "Stanford at TAC KBP 2016: Sealing Pipeline Leaks and Understanding Chinese." In *Proceedings of Text Analysis Conferenc*e, 2016.

7 LIST OF SYMBOLS, ABBREVIATIONS AND ACRONYMS

AAAI	(conference of the) Association for the Advancement of Artificial Intelligence
ACL	(conference of the) Association for Computational Linguistics
BLP	Bayesian Logic Program
COLING	International conference on Computational Linguistics
CONLL	Conference on Natural Language Learning
EACL	(conference of the) European Chapter of the ACL
IJCAI	International Joint Conference on Artificial Intelligence
IWCS	International Conference on Computational Semantics
KBP	Knowledge Base Population
LSTM	Long Short-Term Memory
MLN	Markov Logic Network
NAACL	(conference of the) North American Chapter of the ACL
PPDB	Paraphrase database
PSL	Probabilistic Soft Logic
RTE	Recognizing Textual Entailment
SRL	Statistical Relational Learning
STS	Semantic Textual Similarity
SWAF	Stacking With Additional Features